- ## C++ Introduction
  String, integers, namespaces. Declaration and initialization. Assignment and increment. Conversions. Arrays. Matrices. Structs. Declarations and definitions. Vectors.

- ## Classes
  Classes and objects. Headers and source files. Constructors and destructor. Structs vs classes. "this", "const". Helper functions. Operators: overloading, member functions vs non-member functions. Friends. Static class members.

- ## Pointers (/Smart Pointers), References, Functions Parameters, Iterators
  Pointers. Variables and memory. Dereferencing a pointer and memory leak. Parameters passing. References. "const". Functions parameters: overloading. "auto". Traversing a vector. Iterators. Smart pointers: shared pointers.

- ## Inheritance and Polymorphism
  Inheritance. Base class and derived classe: what is hinerited. Public, private and protected. Constructors and destructor. Polymorphism. Overwriting methods: overloading, redefinition and overriding. Dynamic binding. Abstrac base classes and pure virtual functions. Derived-to-base conversion. Static and dynamic type. Virtual functions.
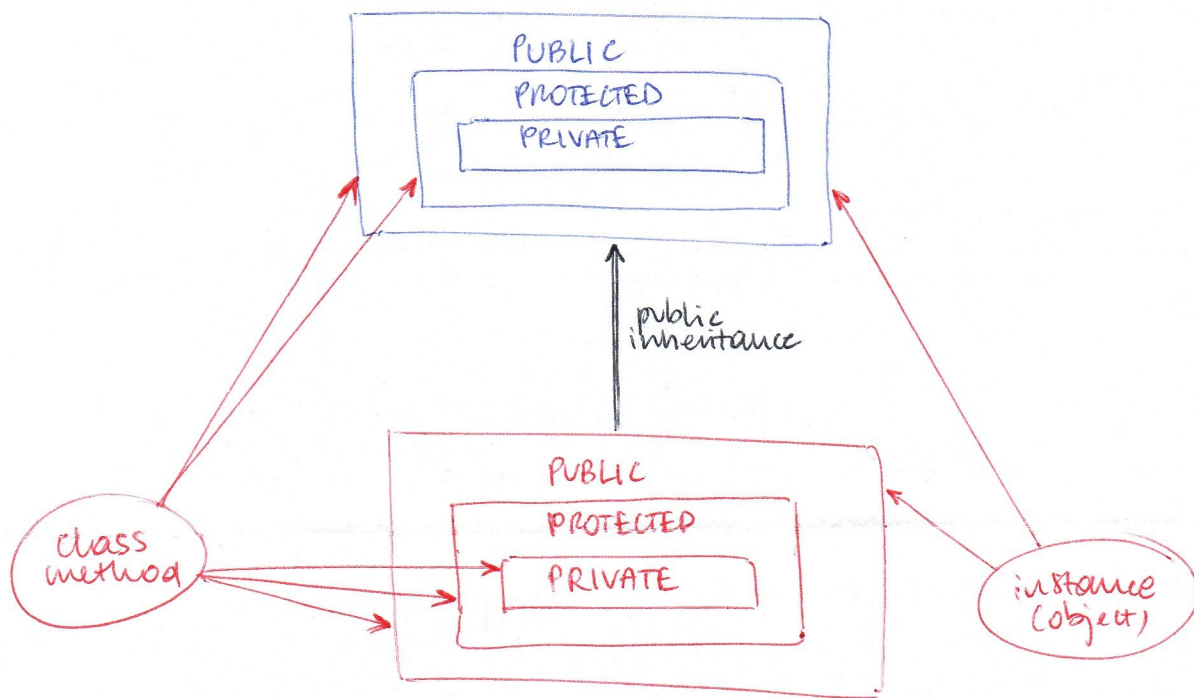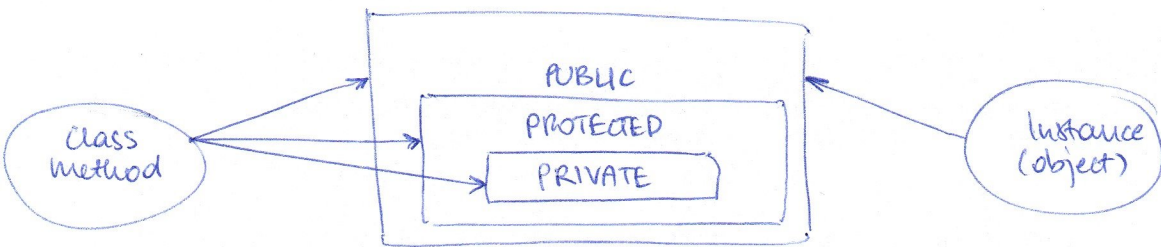
- ## Copy Control
  Constructors. Copy, assignment and destruction. Delete. Like-a-value vs like-a-pointer. Implicit class-type conversions: "explicit".
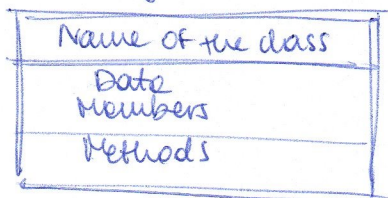
- ## Standard Template Library
  Containers. Sequential containers. Vectors: reserve, resize, push_back. List, forward_list. Deque. Containers common types and operations. Associative containers. Map, set, pair type. Unordered associative containers.

- ## MPI
  Shared and distributed memory. Amdahl's law. What slows down. MPI_Init and MPI_Finalize. Rank and size. MPI_Send and MPI_Recv. Message passing and related problems. MPI_Bcast. MPI_Reduce and MPI_Allreduce. Scatter (block partitioning) and gather. Read_vector and print_vector.

Diagram 1 (top)

```
          PUBLIC
        ┌──────────────────┐
 class  │  PROTECTED       │  Instance
 method │  ┌────────────┐  │  (object)
 ───────│  │ PRIVATE    │  │ ◄──────
        │  └────────────┘  │
        └──────────────────┘
```

class method ──► [PUBLIC / PROTECTED / PRIVATE] ◄── Instance (object)

Diagram 2 (middle)

PUBLIC
PROTECTED
PRIVATE

↑ public
  inheritance

class method

PUBLIC
PROTECTED
PRIVATE

instance (object)

Class diagrams (syntetic repr.)

class diagrams (syntetic repr.)

Name of the class
Date Members
Methods

"+" public
"-" private
"#" protected

e.g.

library
- books []
:

to say that they have a collection